



Application des Champs Conditionnels Aléatoires à l'étiquetage de flux télévisuel

Emmanuelle Martienne, Vincent Claveau, Patrick Gros

► To cite this version:

Emmanuelle Martienne, Vincent Claveau, Patrick Gros. Application des Champs Conditionnels Aléatoires à l'étiquetage de flux télévisuel. RFIA - Reconnaissance des Formes et Intelligence Artificielle, Jan 2012, Lyon, France. pp.978-2-9539515-2-3. hal-00656547

HAL Id: hal-00656547

<https://hal.science/hal-00656547>

Submitted on 17 Jan 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Application des Champs Conditionnels Aléatoires à l'étiquetage de flux télévisuel

Emmanuelle Martienne¹

Vincent Claveau²

Patrick Gros³

¹ Université Rennes 2 - IRISA

² CNRS - IRISA

³ INRIA - Centre Rennes Bretagne Atlantique

Campus de Villejean - 35043 Rennes cedex
Emmanuelle.Martienne@univ-rennes2.fr

Résumé

Nous nous intéressons à la structuration de flux télévisuels et nous abordons plus particulièrement le problème de l'étiquetage : une fois les segments identifiés, il s'agit de leur attribuer à chacun le type correspondant à son contenu sémantique. Pour réaliser cette tâche, différentes techniques de classification et d'apprentissage ont été utilisées : Programmation Logique Inductive, Machines à Vecteurs de Support, Arbres de décision, etc. Dans cet article, nous proposons d'étiqueter les segments à l'aide des Champs Conditionnels Aléatoires, un outil efficace pour l'étiquetage de données séquentielles dans de nombreux domaines. Pour montrer l'intérêt de cette approche, nous présentons diverses expérimentations, réalisées sur des flux segmentés manuellement et automatiquement. Nous faisons également varier la granularité de l'étiquetage et les descripteurs caractérisant les segments. Les résultats obtenus montrent que notre approche est plus robuste que les autres méthodes de classification, en particulier grâce à la prise en compte du voisinage d'un segment dans la détermination de son type. De plus, nous mettons en évidence l'influence du type de segmentation et du choix des descripteurs des segments dans la qualité de l'étiquetage.

Mots Clef

Structuration de flux vidéo, Champs Conditionnels Aléatoires.

Abstract

In this paper, we consider the issue of structuring large TV streams. More precisely, we focus on the labeling problem : once segments have been extracted from the stream, the problem is to automatically label them according to their type (eg. programs vs. commercial breaks). In the literature, several machine learning techniques have been proposed to solve this problem : Inductive Logic Programming, numeric classifiers like SVM or decision trees... In this paper, we propose to use a very effective approach based on another classifier : the Conditional Random Fields (CRF), a tool which has proved useful to handle sequen-

tial data in other domains. We report different experiments, conducted on some manually and automatically segmented data, with different label granularities and different features to describe segments. We demonstrate that this approach is more robust than other classification methods, in particular when it uses the neighbouring context of a segment to find its type. Moreover, we highlight that the segmentation and the choice of features to describe segments are two crucial points in the labeling process.

Keywords

TV stream structuring, Conditional Random Fields.

1 Introduction

De nombreuses chaînes de télévision numérique ont fait leur apparition ces dernières années et de larges volumes de données vidéo sont disponibles. Les chaînes cherchent désormais à développer de nouveaux produits et services tels que la télévision à la demande, la télévision de rattrapage ou les moteurs de recherche vidéo. Qu'il s'agisse de vidéo à la demande ou d'archivage de programmes télévisés, une même nécessité s'impose à toutes les applications exploitant des flux vidéo : structurer ces flux de manière à pouvoir accéder individuellement aux différents programmes qui les composent. La structuration manuelle de ces flux est beaucoup trop coûteuse pour la plupart des applications. De plus, l'exploitation du guide de programme électronique diffusé avec les flux est également peu envisageable, ce guide étant généralement très imprécis [6] : les inter-programmes et les programmes courts n'y sont pas répertoriés, et les heures de début des programmes ne sont que des estimations trop peu précises. Il existe donc un réel besoin de développement d'outils de structuration automatique. Le développement de tels outils est un problème qui est loin d'être trivial. En effet, si d'un point de vue humain un flux vidéo est facilement décomposable en une suite de programmes séparés par des coupures diverses (les inter-programmes), l'examen du signal vidéo ne révèle qu'une suite d'images sans structure apparente. Cette absence de structure au niveau du signal complique fortement la mise

au point de méthodes automatiques de structuration. Généralement, la structuration automatique d'un flux s'effectue en deux étapes : (1) la segmentation, consistant à déterminer précisément les bornes des différents segments du flux ; (2) l'étiquetage, consistant à reconnaître le type de chaque segment (programme, spot de publicité, bande-annonce, etc). Dans cet article, nous nous intéressons à la seconde phase d'étiquetage et plus particulièrement à l'étiquetage par classification. Nous supposons que les segments vidéo du flux ont été identifiés soit manuellement, soit en utilisant une méthode existante [4, 7, 15], et nous essayons de les classer en programmes et inter-programmes, mais également en différents types d'inter-programmes (publicité, bande-annonce, etc). Différentes méthodes de classification ont déjà été utilisées pour effectuer cette tâche : des classifieurs symboliques tels que la Programmation Logique Inductive mais également des classifieurs numériques tels que les modèles de Markov et les SVM (cf. paragraphe 2). Leur principal inconvénient est le manque de robustesse puisqu'elles ont été testées uniquement sur des flux vidéo aux contenus stables et structurés, ce qui est pourtant rarement le cas, en particulier lorsque les flux contiennent plusieurs mois de diffusion. Dans l'approche que nous proposons, nous assimilons le problème de la recherche du type d'un segment à celui de la recherche de l'étiquette associée à un élément particulier dans une séquence de données. Nous faisons également l'hypothèse que les différents segments d'un flux obéissent à des schémas de répétition et que le type d'un segment dépend de ceux de ses proches voisins. Nous utilisons alors un outil qui nous semble plus adapté à la tâche d'étiquetage : les Champs Conditionnels Aléatoires (CCA, en anglais Conditional Random Fields). L'objectif des CCA est d'inférer la séquence d'étiquettes associée à une séquence d'observations, en tenant compte pour chaque observation des étiquettes et observations qui la précèdent et la suivent. Nous réalisons des expérimentations sur un flux vidéo contenant trois semaines de diffusion, et nous montrons que les résultats obtenus par les CCA sont supérieurs à ceux des autres méthodes de classification. Le paragraphe 2 est un aperçu des méthodes de structuration de flux télévisuel existantes. Dans le paragraphe 3, nous présentons les jeux de données sur lesquels nos expérimentations ont été réalisées. Puis, le paragraphe 4 présente l'approche mise en oeuvre pour évaluer la pertinence des CCA dans l'étiquetage des segments. Les paragraphes 5 et 6 comparent les résultats obtenus par cette approche avec ceux des autres méthodes de classification. Nous concluons dans le paragraphe 7 en donnant quelques orientations pour nos futurs travaux.

2 Travaux antérieurs

Il existe deux manières d'aborder le problème de la structuration de flux vidéo : soit de manière descendante (*top-down*), c'est-à-dire en partant d'une information de haut niveau pour aller vers le signal, soit de manière ascendante (*bottom-up*), c'est-à-dire en partant du signal pour

aller vers du contenu sémantique. Poli et al. [9] sont à notre connaissance les seuls à avoir développé une méthode descendante. Ils partent de l'hypothèse que la structure des flux est stable sur une très longue période, les chaînes ayant obligation de proposer pendant longtemps les mêmes programmes quotidiens pour fidéliser leurs téléspectateurs. Leur modèle de prédiction, qui est un modèle de Markov caché, est obtenu par apprentissage sur le guide des programmes de la journée courante et une grande base de données comportant une année entière de vérité terrain. Ce modèle est ensuite utilisé pour déterminer les bornes des programmes ainsi que leurs types (programmes ou inter-programmes) dans le flux. Cette méthode a obtenu de très bons résultats en terme de précision de la prédiction. Son principal inconvénient est qu'elle n'est pas adaptée lorsque les flux contiennent des programmes différents des grilles habituelles, comme par exemple des programmes diffusés uniquement en période de vacances ou des événements sportifs (Jeux olympiques, Tour de France, etc). De plus, elle nécessite énormément de données d'apprentissage annotées manuellement. La plupart des méthodes sont ascendantes et structurent un flux uniquement en y détectant les publicités, à partir des caractéristiques audio et visuelles du signal. La première méthode qui s'intéresse entièrement au problème de la structuration est celle développée par Natu-rel [7]. Elle repose principalement sur l'utilisation d'un ensemble de référence dans lequel les programmes de la première journée du flux ont été annotés manuellement. Cet ensemble de référence est ensuite utilisé pour détecter les programmes qui se répètent et structurer petit à petit la partie restante du flux. Les programmes non répétés ne figurant pas dans l'ensemble de référence sont étiquetés grâce à des post-traitements puis ajoutés dans l'ensemble de référence afin de le maintenir à jour. Le principal inconvénient de cette méthode est l'annotation manuelle de l'ensemble de référence. L'annotation manuelle est une tâche fastidieuse et longue qui est rarement envisageable. De plus, le flux utilisé pour tester l'approche, d'une durée de trois semaines, est trop réduit pour assurer la qualité des résultats sur de très grands flux. Une autre méthode, proposée par Manson et al. [4], consiste à utiliser un système d'apprentissage basé sur la Programmation Logique Inductive pour inférer le type d'un segment à partir de son contexte local. L'inconvénient est que sept jours de données annotées manuellement sont nécessaires pour construire le modèle d'apprentissage. De plus, les tests ont été réalisés sur des flux correspondant à la tranche horaire 18h-0h, connue pour être stable et peu sujette aux changements. La fiabilité de la méthode sur des flux plus instables reste à démontrer. Dans cet article, nous proposons l'utilisation d'une technique plus robuste que celles évoquées précédemment.

3 Données

Le flux vidéo que nous utilisons contient trois semaines complètes de programmes diffusés sur une chaîne publique française. Deux jeux de données ont été extraits de ce flux.

Date	Type	Type IP	Image début	Image fin	Segment
09052005	1	1	1	27258	<i>Seg₅₄</i>
09052005	2	4	27259	27563	<i>Seg₅₅</i>
09052005	2	3	27575	27764	<i>Seg₅₆</i>
09052005	2	2	27767	27834	<i>Seg₅₇</i>
09052005	2	4	27835	28758	<i>Seg₅₈</i>

TAB. 1 – Echantillon de jeu de données

Jeu de données		manuel	automatique
Type			
programme		1506	22557
bande-annonce		1290	4075
publicité		1050	18089
sponsoring		1714	2201
habillage		2031	1622
total		7591	48544

TAB. 2 – Répartition des différents types de segments au sein des jeux de données

Le premier, que nous appellerons *le jeu de données manuel*, est le résultat d'une segmentation manuelle du flux. Il va nous servir à évaluer uniquement le processus d'étiquetage des segments, sans l'influence d'un outil de segmentation automatique. Le second jeu de données, qui sera appelé *le jeu de données automatique*, est obtenu grâce à une technique de segmentation automatique basée sur la détection de répétitions de segments [1]. Il va nous permettre d'évaluer la pertinence de notre approche dans le contexte d'une segmentation automatique. Dans chacun des jeux de données, les segments sont décrits à l'aide de leurs dates de diffusion et de leurs images de début et de fin dans le flux. Une étiquette précise également pour chaque segment s'il est un programme ou un inter-programme (IP). Dans le cas d'un IP, une étiquette supplémentaire permet de préciser son type exact : *publicité*, *bande-annonce*, *sponsoring* ou *habillage*. Le tableau 1 présente un échantillon de jeu de données. Les segments y apparaissent dans l'ordre chronologique à raison d'un segment par ligne. Dans la colonne **Type**, les valeurs 1 et 2 correspondent respectivement à un programme et un IP. La colonne **Type IP** fournit un typage plus précis des IPs : 1 pour une publicité, 2 pour un habillage, 3 pour un sponsoring et 4 pour une bande-annonce. Les segments correspondant à des programmes ont tous la même valeur 1 dans la colonne **Type IP**.

Le tableau 2 montre pour chaque jeu de données le nombre total de segments qu'il contient ainsi que la répartition des segments entre les différents types. Nous pouvons remarquer que le jeu de données automatique contient beaucoup plus de segments puisque l'usage d'une méthode de segmentation automatique conduit très souvent à une sur-segmentation du flux initial.

Toutes les expérimentations présentées dans cet article ont été réalisées sur la base de ces deux jeux de données. Deux types de descripteurs ont été extraits pour décrire les segments : les descripteurs robustes et les descripteurs de contenu.

Segment	Moment de la semaine	Période du jour	Durée	Type
<i>Seg₃₃</i>	jour ouvré	matin	[0,10s[publicité
<i>Seg₃₄</i>	jour ouvré	matin	[30s,1m[habillage
<i>Seg₃₅</i>	jour ouvré	matin	[5m,+[programme
<i>Seg₃₆</i>	jour ouvré	midi	[0,10s[habillage
<i>Seg₃₇</i>	jour ouvré	midi	[0,10s[bande-annonce

TAB. 3 – Séquence de segments décrits à l'aide des descripteurs robustes

3.1 Descripteurs robustes

Les descripteurs robustes doivent se calculer très facilement et ne doivent pas dépendre de la qualité de signal du flux. Dans notre cas, ces descripteurs incluent la durée du segment, sous la forme d'un intervalle, ainsi que deux autres propriétés liées à la diffusion telles que le moment de la semaine (jour ouvré, week-end ou vacances) et la période de la journée (nuit, matin, midi, etc). Le tableau 3 montre un exemple de séquence décrite à l'aide des descripteurs robustes.

3.2 Descripteurs de contenu

Les descripteurs de contenu sont plus liés aux propriétés que possèdent les segments au sein du flux télévisuel. Dans notre cas, ils peuvent être divisés en deux catégories : les descripteurs *globaux* et les descripteurs *locaux*. La première catégorie comprend le nombre d'occurrences d'un segment (un même segment pouvant figurer à plusieurs reprises dans un flux), sa durée, le nombre de jours calendaires et le nombre de jours différents de la semaine où il a été diffusé¹. Dans la seconde catégorie, les descripteurs locaux d'un segment sont extraits à partir de deux sources d'information : (1) la présence d'une séparation avant et/ou après ce segment² et (2) ses segments avoisinants. Soient W_b et W_a les deux fenêtres temporelles situées respectivement avant et après le segment. Les descripteurs issus de ces fenêtres temporelles sont :

- les nombres de segments appartenant à W_b et W_a ;
- les nombres totaux d'occurrences des segments appartenant à W_b et W_a ;
- les nombres minimaux, maximaux et moyens d'occurrences des segments appartenant à W_b et W_a .

Au total, 15 descripteurs de contenu sont utilisés pour caractériser un segment.

4 Principe de l'étiquetage avec les CCA

4.1 Champs Conditionnels Aléatoires

L'objectif des Champs Conditionnels Aléatoires (CCA) [3] est de construire des modèles probabilistes capables d'an-

¹Par exemple, pour un segment diffusé durant 10 mardis consécutifs le nombre de jours calendaires sera de 10 alors que le nombre de jours différents de la semaine sera de 1.

²Une séparation désigne les occurrences simultanées d'images monochromes et de silence se produisant entre les publicités à la télévision française. Ces séparations sont imposées par la législation française.

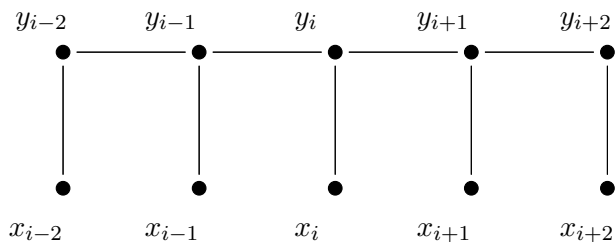


FIG. 1 – Structure graphique des CCA pour les séquences

noter des données. Étant donnée une observation x , un modèle produit par les CCA doit permettre de déterminer l'étiquette qui lui est associée y , à partir d'un ensemble d'observations déjà annotées, c'est-à-dire à partir d'un ensemble de paires (x, y) , x et y pouvant être de natures très variées. Les CCA ont dans un premier temps été utilisés avec succès pour l'étiquetage de données séquentielles, c'est-à-dire pour inférer la séquence d'étiquettes $y = (y_1, \dots, y_n)$ associée à une séquence d'observations $x = (x_1, \dots, x_n)$. Ils sont actuellement très largement employés en traitement des langues naturelles pour des tâches d'annotation d'arbres XML [2] ou d'arbres syntaxiques [5], ainsi que dans le cadre de l'étiquetage grammatical et de l'analyse de texte [8, 11]. En biologie computationnelle, ils ont été appliqués à l'analyse de la structure secondaire de l'ARN [12]. Dans le domaine du multimedia, ils ont permis de résoudre certains problèmes de détection [14, 13, 10]. Plus précisément, les CCA appartiennent à la famille des modèles graphiques (cf. figure 1) dont les Modèles de Markov Cachés (MCC) font également partie. Il existe toutefois une différence majeure entre les deux approches puisque les CCA considèrent la probabilité conditionnelle $P(y|x)$ d'une séquence d'étiquettes y étant donnée une séquence d'observations x , alors que les MMC définissent une probabilité jointe $P(x, y)$. La conséquence est que les MMC font l'hypothèse d'indépendance des observations, mais pas les CCA. Dans les CCA, les probabilités des transitions entre étiquettes peuvent donc dépendre des observations passées et futures et pas uniquement de l'observation courante comme c'est le cas dans les MMC (une étiquette y_i est reliée à tous les x_j). L'usage des CCA paraît donc tout à fait adapté pour traiter des séquences de segments, en particulier parce qu'ils permettront de tenir compte des segments avoisinants dans la construction du modèle d'étiquetage.

4.2 Étiquetage de segments à l'aide des CCA

Un flux télévisuel peut-être vu comme une séquence de segments vidéo, où les caractéristiques des segments sont les observations et les types des segments les étiquettes qui leur sont associées. Le problème de l'étiquetage des segments peut donc être assimilé à un problème classique d'étiquetage de séquences que permettent de résoudre les CCA. L'objectif de cet article est d'évaluer la pertinence

des CCA dans le contexte de l'étiquetage des segments vidéo. La question que nous posons est la suivante : les CCA sont-ils efficaces pour prédire les types des segments à partir de leurs caractéristiques ? Les expérimentations que nous menons pour y répondre mettent en oeuvre un processus couramment utilisé dans l'évaluation des techniques d'apprentissage. Une première séquence, appelée *séquence d'apprentissage*, est utilisée par les CCA pour construire le modèle probabiliste. Ce modèle est ensuite testé sur une seconde séquence (différente de la séquence d'apprentissage), appelée *séquence de test*. À l'issue de l'étiquetage de la séquence de test, la comparaison entre les types réels et ceux prédits par le modèle permettent de calculer des mesures, telles que la *précision* et le *rappel*, évaluant la pertinence du modèle produit pour l'étiquetage d'une nouvelle séquence. Les deux séquences sont extraites du même jeu de données (voir paragraphe 3). Les segments y sont décrits avec les mêmes descripteurs, robustes ou de contenu.

5 Expérimentations sur le jeu de données manuel

Les premières expérimentations ont été menées sur le jeu de données manuel, en utilisant les descripteurs robustes. Leur objectif est d'évaluer la qualité de l'étiquetage lorsque la segmentation est parfaite et que les segments sont décrits avec des caractéristiques simples. Les deux premières semaines du flux télévisuel ont été utilisées comme la séquence d'apprentissage, la troisième semaine restante comme la séquence de test. Deux expérimentations ont été faites de manière à faire varier la granularité de l'étiquetage. Dans la première, les segments sont étiquetés soit comme des programmes, soit comme des IP. Dans la seconde, une distinction est faite entre les différents types d'IP. Cinq types sont alors possibles : programme, publicité, bande-annonce, habillage ou sponsoring. Le logiciel open source implémentant les CCA que nous avons utilisé est *CRF++*, développé par Taku Kudo³. Les patrons utilisés dans *CRF++* sont 'unigram' relativement aux étiquettes (il n'est tenu compte que du type du segment courant). En revanche, au niveau des observations les quatre segments antérieurs et les quatre segments postérieurs sont pris en compte. La particularité du processus d'apprentissage mis en oeuvre par les CCA est qu'il tient compte de la sequentialité des données qu'il traite. Pour montrer que la prise en compte de cet aspect des données est essentiel dans le processus d'étiquetage des segments, il nous faut comparer les résultats obtenus par les CCA avec ceux obtenus par d'autres méthodes de classification. Par conséquent, des expérimentations similaires ont été menées avec d'autres méthodes sur le même jeu de données manuel. Les tableaux 4 et 5 montrent l'ensemble des résultats obtenus. Seuls les résultats des trois meilleures méthodes de classification sont donnés.

Nous pouvons tout d'abord remarquer que les résultats des

³<http://crfpp.sourceforge.net/>

Méthode	Type	Précision	Rappel	F-mesure
CCA	IP	95.49%	97.80%	96.63%
	Programme	90.61%	82.09%	86.14%
KNN	IP	91.92%	95.28%	93.57%
	Programme	78.72%	67.55%	72.71%
C4.5	IP	94.63%	88.69%	91.56%
	Programme	64.76%	80.50%	71.78%
Arbre de décision	IP	94.21%	89.33%	91.71%
	Programme	65.58%	78.72%	71.55%

TAB. 4 – Résultats des CCA et d’autres méthodes de classification - Étiquetage programme vs. IP

Méthode	Type	Précision	Rappel	F-mesure
CCA	Programme	88.43%	88.12%	88.27%
	Bande-annonce	88.33%	87.40%	87.86%
	Habillage	78.27%	82.85%	80.49%
	Publicité	84.51%	81.23%	82.84%
	Sponsoring	79.89%	77.09%	78.47%
KNN (10NN)	Programme	78.76%	67.73%	72.83%
	Bande-annonce	93.27%	72.70%	81.71%
	Habillage	50.08%	86.57%	63.45%
	Publicité	50.36%	66.99%	57.50%
	Sponsoring	59.22%	20.35%	30.29%
Bayésienne naïve	Programme	67.79%	64.18%	65.94%
	Bande-annonce	93.02%	83.99%	88.27%
	Habillage	50.60%	89.10%	64.54%
	Publicité	44.03%	45.31%	44.66%
	Sponsoring	63.60%	19.54%	29.90%
SVM	Programme	65.58%	78.72%	71.55%
	Bande-annonce	93.02%	83.99%	88.27%
	Habillage	50.60%	89.10%	64.54%
	Publicité	44.57%	25.24%	32.23%
	Sponsoring	63.60%	19.54%	29.90%

TAB. 5 – Résultats des CCA et d’autres méthodes de classification - Étiquetage multiple

CCA sont supérieurs à ceux des autres méthodes, plus particulièrement lorsque l’étiquetage de la séquence de test doit séparer les différents types d’IP. La prise en compte de la séquentialité semble donc améliorer les performances du modèle d’étiquetage. Dans le cas d’un étiquetage multiple, les précisions et rappels du modèle des CCA varient d’un type de segment à l’autre tout en restant relativement élevés (de 78.27% à 88.43% pour la précision et de 77.09% à 88.12% pour le rappel). Néanmoins, le modèle a plus de difficultés pour séparer les différents types de segments puisque ces mêmes mesures sont plus élevées lorsque l’étiquetage est binaire : programme vs. IP. Sur la matrice de confusion (voir tableau 6), nous voyons que le modèle fait souvent la confusion entre les types habillage et sponsoring : 125 segments de type habillage sont reconnus comme de type sponsoring, inversement 151 segments de type sponsoring sont reconnus comme de type habillage. Cette confusion est également observée entre les types publicité et programme. Cette baisse d’efficacité du modèle en cas d’étiquetage multiple peut être attribuée à une faiblesse des descripteurs robustes, suffisants pour séparer les programmes des IPs mais insuffisants pour faire une distinction plus fine entre des segments IP de types différents mais ayant des caractéristiques proches.

		Réal				
		Habillage	Spons.	Bande-an.	Pub.	Prog.
Prédit	Habillage	623	151	21	0	1
	Spons.	125	572	14	1	4
	Bande-an.	2	13	333	6	23
	Pub.	0	2	5	251	39
	Prog.	2	4	8	51	497

TAB. 6 – Matrice de confusion du modèle pour chaque type de segment

Méthode	Type	Précision	Rappel	F-mesure
CCA	IP	64.97%	90.34%	75.58%
	Programme	81.63%	46.83%	59.52%
SVM	IP	58.65%	72.41%	64.81%
	Programme	59.52%	44.28%	50.78%
CN2	IP	59.05%	62.03%	60.50%
	Programme	56.14%	53.04%	54.55%
Arbre de décision	IP	59.09%	61.99%	60.51%
	Programme	56.16%	53.16%	54.62%

TAB. 7 – Résultats des CCA et d’autres méthodes de classification - Étiquetage programme vs. IP

6 Expérimentations sur le jeu de données automatique

Les expérimentations sur le jeu de données manuel ont montré un intérêt réel à l’utilisation des CCA pour étiqueter les segments lorsque la segmentation est parfaite. Cependant, segmenter manuellement les flux vidéo n’est pas une approche réaliste. L’objectif de ces nouvelles expérimentations, réalisées dans le même contexte que les précédentes (voir paragraphe 5), est d’évaluer les CCA sur des séquences obtenues par segmentation automatique. Nous utilisons donc le jeu de données automatique, dans lequel les segments sont décrits à l’aide des descripteurs robustes. Les tableaux 7 et 8 présentent les résultats obtenus par les CCA et les autres méthodes de classification.

Dans le paragraphe 3, nous avons mis en évidence le nombre élevé de segments figurant dans le jeu de données automatique dû à l’utilisation d’une méthode de segmentation automatique. Cette sur-segmentation semble avoir un réel impact sur les performances de l’étiquetage. En effet, toutes les méthodes sont moins efficaces pour distinguer les programmes des IP. Bien que les CCA obtiennent encore les meilleurs résultats, le nombre de segments consécutifs ayant des caractéristiques très proches est si élevé qu’il est difficile de prédire leurs types réels. Les résultats sont encore plus mauvais lorsque l’étiquetage est multiple : les types habillage et sponsoring ne sont quasiment plus identifiés, les précisions et rappels sont très bas sur la plupart des types d’IP.

Ces expérimentations ainsi que celles décrites dans le paragraphe 5 montrent que les descripteurs robustes sont efficaces dans le cas d’une segmentation manuelle, mais sont insuffisants pour distinguer les différents types de segments lorsque la méthode de segmentation utilisée est automa-

Méthode	Type	Précision	Rappel	F-mesure
CCA	Programme	67.17%	64.61%	65.87%
	Bande-annonce	10.17%	1.23%	2.19%
	Habillage	0%	0%	0%
	Publicité	49.26%	74.37%	59.27%
	Sponsoring	20.45%	0.94%	1.80%
Arbre de décision	Programme	51.81%	69.64%	59.42%
	Bande-annonce	25.93%	0.48%	0.94%
	Habillage	0%	0%	0%
	Publicité	39.94%	40.72%	40.33%
	Sponsoring	0%	0%	0%
CN2	Programme	51.78%	69.78%	59.45%
	Bande-annonce	26.47%	0.62%	1.21%
	Habillage	0%	0%	0%
	Publicité	39.87%	40.41%	40.14%
	Sponsoring	0%	0%	0%
C4.5	Programme	51.80%	72.02%	60.26%
	Bande-annonce	0%	0%	0%
	Habillage	0%	0%	0%
	Publicité	39.96%	38.40%	39.16%
	Sponsoring	0%	0%	0%

TAB. 8 – Résultats des CCA et d’autres méthodes de classification - Étiquetage multiple

Méthode	Type	Précision	Rappel	F-mesure
CCA	IP	93.72%	94.76%	94.24%
	Programme	93.26%	91.95%	92.60%
SVM	IP	95.07%	91.92%	93.47%
	Programme	90.17%	93.96%	92.03%
CN2	IP	94.78%	91.97%	93.35%
	Programme	90.19%	93.58%	91.85%
Bayésienne naïve	IP	91.52%	94.87%	93.16%
	Programme	93.18%	88.86%	90.97%

TAB. 9 – Résultats des CCA et d’autres méthodes de classification - Étiquetage programme vs. IP

tique. Cela signifie que le gain de temps de calcul réalisé par l’utilisation de ces descripteurs est perdu par la nécessité de segmenter manuellement le flux. L’objectif de ces dernières expérimentations est de voir si l’usage d’autres descripteurs, en particulier des descripteurs de contenu, se révèle plus pertinente sur les séquences segmentées automatiquement. Le jeu de données utilisé est le jeu de données automatique, dans lequel les segments sont décrits à l’aide des descripteurs de contenu (voir paragraphe 3.2) et identifiés soit comme programmes soit comme IP. Le jeu de données étant très volumineux, 30% des segments ont été utilisés comme séquence d’apprentissage, les 70% restant comme séquence de test⁴. Le tableau 9 suivant montre les résultats obtenus.

Nous voyons que l’utilisation des descripteurs de contenu permet d’augmenter significativement les résultats de tous les classificateurs, alors même que la séquence d’apprentissage contient moins de segments que dans les expérimentations précédentes. L’utilisation d’une méthode de segmentation automatique nécessite donc l’emploi de descripteurs liés au contenu dont le calcul est loin d’être trivial.

⁴Les deux séquences ont été extraites tout en préservant la séquentialité

7 Conclusion et perspectives

Dans cet article, nous avons montré que les Champs Conditionnels Aléatoires étaient particulièrement adaptés à la tâche d’étiquetage des segments vidéo. Les expérimentations réalisées ont montré que les résultats obtenus par les CCA sont supérieurs à ceux d’autres méthodes de classification. Elles ont mis en évidence la robustesse de cette méthode pour séparer les programmes des inter-programmes, mais également pour identifier les différents types d’inter-programmes lorsque les segments résultent d’une segmentation manuelle ou automatique. Néanmoins, le choix des descripteurs de segments est important : les descripteurs robustes, simples à calculer, se révèlent insuffisants pour identifier les segments lorsque la segmentation est automatique. Dans ce cas, l’utilisation de descripteurs de contenus plus complexes à obtenir est indispensable. Pour disposer d’un modèle d’étiquetage efficace, il est nécessaire de consacrer beaucoup de temps, soit au niveau de la segmentation, soit au niveau du calcul des descripteurs. La principale limite des CCA est qu’il s’agit d’une méthode supervisée, c’est-à-dire d’une méthode qui construit son modèle à partir de données déjà étiquetées. Étiqueter une partie des segments manuellement est un travail fastidieux. Nos futurs travaux seront principalement orientés vers la réduction du besoin de données d’apprentissage. En particulier, nous souhaitons étudier l’utilisation des CCA dans un contexte d’apprentissage non supervisé, en nous inspirant de travaux similaires menés sur d’autres méthodes d’apprentissage.

Références

- [1] Z. A. A. Ibrahim and P. Gros. Tv stream structuring. *ISRN Signal Processing*, 2011.
- [2] F. Jousse. *Transformation d’arbres XML avec des modèles probabilistes pour l’annotation*. PhD thesis, University of Lille III, France, 2007.
- [3] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proc. of the Int. Conf. on Machine Learning (ICML)*, pages 282–289, July 2001.
- [4] G. Manson and S.-A. Berrani. An inductive logic programming-based approach for TV stream segment classification. In *Proc. of the IEEE Int. Symp. on Multimedia*, December 2008.
- [5] B. A. L. G. R. A. Moreau E., Tellier I. and P. T. Annotation fonctionnelle de corpus arborés avec des champs aléatoires conditionnels. In *TALN*.
- [6] X. Naturel. *Structuration automatique de flux vidéos de télévision*. PhD thesis, University of Rennes 1, France, 2007.
- [7] X. Naturel, G. Gravier, and P. Gros. Fast structuring of large streams using program guides. In *Adaptive Multimedia Retrieval*, pages 222–231, July 2006.

- [8] D. Pinto, A. McCallum, X. Wei, and W. Croft. Table extraction using Conditional Random Fields. In *Proc. of the ACM SIGIR*, pages 235–242, July 2003.
- [9] J.-P. Poli. An automatic television stream structuring system for television archives holders. *Multimedia systems*, 14(5) :255–275, November 2008.
- [10] A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In *Neural Information Processing Systems (NIPS)*, December 2004.
- [11] F. Sha and F. Pereira. Shallow parsing with conditional random fields. In *Proc. of Human Language Technology - North American Chapter of the Association for Computational Linguistics*, May 2003.
- [12] Y. Tabei and K. Asai. A local multiple alignment method for detection of non coding RNA sequences. *Bioinformatics*, 25 :1498–1505, 2009.
- [13] J. Weinman, A. Hanson, and A. McCallum. Sign detection in natural images with Conditional Random Fields. In *Proc. of the IEEE International Workshop on Machine Learning for Signal Processing*, September 2004.
- [14] J. Yuan, J. Li, and B. Zhang. Gradual transition detection with conditional random fields. In *Proc. of the 15th International Conference on Multimedia*, pages 277–280. ACM, September 2007.
- [15] Z. Zeng, S. Zhang, H. Zheng, and W. Yang. Program segmentation in a television stream using acoustic cues. In *Proc. of the International Conference on Audio, Language and Image Processing*, pages 748–752, July 2008.